



图文 17 领导的要求：你来对 Kafka、RabbitMQ 以及 RocketMQ 调研

1412 人次阅读 2019-11-01 07:54:52

[详情](#) [评论](#)

领导的要求：

你来对 Kafka、RabbitMQ 以及 RocketMQ 进行技术选型调研

石杉老哥重磅力作：《互联网java工程师面试突击》（第3季）【强烈推荐】：



全程真题驱动，精研Java面试中6大专题的高频考点，从面试官的角度剖析面试

(点击下方蓝字试听)

[《互联网Java工程师面试突击》（第3季）](#)

1、小猛的顿悟：订单系统的很多问题都可以用MQ来解决

小猛昨天听了明哥对MQ这个东西的一番解释之后，已经有了一个基本的概念了，他回到家里仔细的想了想，发现似乎订单系统面临的很多问题都可以用MQ来解决！

举个例子，支付订单流程中步骤过多，导致性能很差。这个问题可以用MQ来解决，让订单系统仅仅完成最核心的一些步骤和调用，然后发送消息到MQ，比如仓储系统之类的就可以从MQ里获取消息，然后慢慢的执行一些很耗时的步骤。

还有比如订单系统在退款的时候，可能会遇到第三方支付系统退款失败的问题，从而影响整个退款流程的失败，这个问题也可以用MQ解决。可以让订单系统在退款的时候完成公司内部各个系统的流程，然后发送消息到MQ，由一个专门的服务去负责调用第三方支付系统，处理可能出现的失败。

在双11大促活动的时候，也可以让瞬间涌入的大量下单请求到MQ里去排队，然后让订单系统在后台慢慢的获取订单，以数据库可以接受的速率完成操作，避免瞬间请求量过大击垮数据库。

小猛想着觉得很兴奋，订单系统的诸多问题似乎都可以基于MQ来解决啊！

他一大早去公司上班之后，就把自己的那些想法告诉了明哥，明哥非常的高兴，他觉得小猛真是太棒了，自己就领悟出来了MQ在订单系统里的各种用法！

2、万事开头难：要在订单系统里用MQ技术，从哪一步开始呢？

听着明哥的表扬，小猛觉得特别的开心，但是另外一方面，他也有一个很大的疑问，虽然知道MQ是一个东西，可以解决订单系统的一些问题，但是万事开头难，到底应该从哪里入手呢？

他跟明哥提出了自己的疑惑，明哥笑着说，要做一些技术架构的升级，引入一些新的技术，当然要从技术调研开始了。

小猛很疑惑：技术调研应该怎么做？

明哥解释道：其实技术调研说白了，就是对一个技术去找到一些业内常用的开源实现，然后对各种不同的实现都进行一些调研，对比一下他们的优劣势，看看谁比较符合我们的需求，谁比较适合我们来使用。

具体来说，比如对于我们现在的情况，你只知道有一个MQ的概念，但是你要考虑一下：

业内常用的MQ有哪些？

每一种MQ各自的表现如何？

这些MQ在同等机器条件下，能抗多少QPS（每秒抗几千QPS还是几万QPS）？

性能有多高（发送一条消息给他要2ms还是20ms）？

可用性能不能得到保证（要是MQ部署的机器挂了怎么办）？

然后你还得考虑：

他们会不会丢失数据？

如果需要的话能否让他们进行线性的集群扩容（就是多加几台机器）？

消息中间件经常需要使用的一些功能他们都有吗（比如说延迟消息、事务消息、消息堆积、消息回溯、死信队列，等等）？

另外还得考虑这些MQ在文档是否齐全？社区是否活跃？在行业内是否广泛运用？是用什么语言编写的？

把这些事情都搞清楚了，那么你就完成了技术调研，可以全面的对比各种MQ的优劣势，然后从中选择一个最适合我们的来使用。

小猛都没反应过来呢，明哥接着说道，这个技术调研的任务，我就打算交给你了！正好让你借着这个机会对各种MQ有一个基本的了解。其实也没你想的那么难，对你来说要做调研的话，就是上网搜集资料，然后进行梳理对比就可以了。

小猛尴尬的说，好吧，那我尽力而为。

3、Kafka、RabbitMQ以及RocketMQ的调研对比

过了几天，小猛就完成了一份MQ技术调研报告，他找到了明哥，开始讲起了自己做这份调研报告的心得体会。

首先，小猛通过网上搜集资料，发现一般国内常用的MQ技术有四种实现，ActiveMQ、Kafka、RabbitMQ、RocketMQ，但是其中ActiveMQ主要是几年以前较多公司使用，现在几乎国内用的公司都很少了。

因此小猛搜集资料的时候主要是针对Kafka、RabbitMQ、RocketMQ三种技术做的调研。

(1) Kafka的优势和劣势

先来说Kafka，小猛通过查阅一些Kafka的基本资料发现，首先Kafka的吞吐量几乎是行业里最优秀的，在常规的机器配置下，一台机器可以达到每秒十几万的QPS，相当的强悍。

Kafka性能也很高，基本上发送消息给Kafka都是毫秒级的性能。可用性也很高，Kafka是可以支持集群部署的，其中部分机器宕机是可以继续运行的。

但是Kafka比较为人诟病的一点，似乎是丢数据方面的问题，因为Kafka收到消息之后会写入一个磁盘缓冲区里，并没有直接落地到物理磁盘上去，所以要是机器本身故障了，可能会导致磁盘缓冲区里的数据丢失。

而且Kafka另外一个比较大的缺点，就是功能非常的单一，主要是支持发送消息给他，然后从里面消费消息，其他就没有什么额外的高级功能了。所以基于Kafka有限的功能，可能适用的场景并不是很多。

因此综上所述，以及查阅了Kafka技术在各大公司里的使用，基本行业里的一个标准，是把Kafka用在用户行为日志的采集和传输上，比如大数据团队要收集APP上用户的一些行为日志，这种日志就是用Kafka来收集和传输的。

因为那种日志适当丢失数据是没有关系的，而且一般量特别大，要求吞吐量要高，一般就是收发消息，不需要太多的高级功能，所以Kafka是非常适合这种场景的。

(2) RabbitMQ的优势和历史

再说RabbitMQ，在RocketMQ出现之前，国内大部分公司都从ActiveMQ切换到RabbitMQ来使用，包括很多一线互联网大厂，而且直到现在都有很多中小型公司在用RabbitMQ。

RabbitMQ的优势在于可以保证数据不丢失，也能保证高可用性，即集群部署的时候部分机器宕机可以继续运行，然后支持部分高级功能，比如说死信队列，消息重试之类的，这些是他的优点。

但是他也有一些缺点，最为人诟病的，就是RabbitMQ的吞吐量是比较低的，一般就是每秒几万的级别，所以如果遇到特别特别高并发的情况下，支撑起来是有点困难的。

而且他进行集群扩展的时候（也就是加机器部署），还比较麻烦。

另外还有一个较为致命的缺陷，就是他的开发语言是erlang，国内很少有精通erlang语言的工程师，因此也没办法去阅读他的源代码，甚至修改他的源代码。

所以现在行业里的一个情况是，很多BAT等一线互联网大厂都切换到使用更加优秀的RocketMQ了，但是很多中小型公司觉得RabbitMQ基本可以满足自己的需求还在继续使用中，因为中小型公司并不需要特别高的吞吐量，RabbitMQ已经足以满足他们的需求了，而且也不需要部署特别大规模的集群，也没必要去阅读和修改RabbitMQ的源码。

(3) RocketMQ的优势和劣势

RocketMQ是阿里开源的消息中间件，久经沙场，非常的靠谱。他几乎同时解决了Kafka和RabbitMQ的缺陷。

RocketMQ的吞吐量也同样很高，单机可以达到10万QPS以上，而且可以保证高可用性，性能很高，而且支持通过配置保证数据绝对不丢失，可以部署大规模的集群，还支持各种高级的功能，比如说延迟消息、事务消息、消息回溯、死信队列、消息积压，等等。

而且RocketMQ是基于Java开发的，符合国内大多数公司的技术栈，很容易就可以阅读他的源码，甚至是修改他的源码。

所以国内很多一线互联网大厂都切换到使用RocketMQ了，他们需要RocketMQ的高吞吐量，大规模集群部署能力，以及各种高阶的功能去支撑自己的各种业务场景，同时还可以根据自己的需求定制修改RocketMQ的源码。

RocketMQ是非常适合用在Java业务系统架构中的，因为他很高的性能表现，还有他的高阶功能的支持，可以让我们解决各种业务问题。

当然，RocketMQ也有一点美中不足的地方，就是经过我的调查发现，RocketMQ的官方文档相对简单一些，但是Kafka和RabbitMQ的官方文档就非常的全面和详细，这可能是RocketMQ目前唯一的缺点。

(4) 活跃的社区和广泛的运用

最后一点，基本上Kafka、RabbitMQ和RocketMQ的社区都还算活跃，更新频率都还可以，而且基本运用都非常的广泛。

尤其是Kafka和RabbitMQ，目前Kafka几乎是国内大数据领域日志采集传输的标准，RabbitMQ在各种中小公司里运用极为广泛，RocketMQ也是开始在一些大公司和其他公司里快速推行中。

4、经验老到的明哥：其实我早有准备

当小猛把这一番调研结果结合自己写好的调研文档给明哥讲完之后，明哥哈哈的笑着说，不错不错，以一个新人的能力而言，做调研到这个程度基本上是差不多了。

不过其实几种MQ的技术对比和分析，我自己也早就做过了，让你去做这个调研，主要是锻炼一下你，让你多了解了解。

小猛有点不好意思的挠挠头说到：原来是这样，那我就不在关公面前耍大刀了，明哥，那你觉得我们该用哪个MQ呢？

似乎看起来RabbitMQ也能满足我们，毕竟我们现在也属于中小型互联网公司，并发量没那么大；但是RocketMQ似乎是更好的选择，他规避掉了RabbitMQ的全部缺点。

明哥说到，虽然我们现在是一个中小型互联网公司，看起来似乎RabbitMQ足以应对我们的需求，但是假设未来我们成长为一个大公司呢？也许我们也会有每秒几十万的QPS，也许我们以后也需要对MQ进行源码的二次开发，那此时RabbitMQ还合适吗？

所以，我的决定是，一步到位，直接用RocketMQ作为我们公司的技术选型。

小猛听完明哥的分析和决定之后，觉得老大霸气侧漏，那就啥也不说了，后面就跟着明哥用RocketMQ去重构订单系统吧！

End

专栏版权归公众号**狸猫技术窝**所有

未经许可不得传播，如有侵权将追究法律责任

狸猫技术窝其他精品专栏推荐：

[《从零开始带你成为JVM实战高手》](#)

[《21天Java 面试突击训练营》（分布式篇）](#)（现更名为：[互联网Java工程师面试突击第2季](#)）

[互联网Java工程师面试突击（第1季）](#)

重要说明：

如何提问：每篇文章都有评论区，大家可以尽情在评论区留言提问，我会逐一答疑

如何加群：购买了狸猫技术窝专栏的小伙伴都可以加入**狸猫技术交流群**

具体加群方式，请参见**目录菜单**下的文档：《付费用户如何加群？》（**购买后可见**）